


SOFTWARE

Open Access



# QuickPIV: Efficient 3D particle image velocimetry software applied to quantifying cellular migration during embryogenesis

Marc Pereyra<sup>1\*†</sup> , Armin Drusko<sup>3†</sup>, Franziska Krämer<sup>2</sup>, Frederic Strobl<sup>2</sup>, Ernst H. K. Stelzer<sup>2</sup> and Franziska Matthäus<sup>1</sup>

\*Correspondence:

pereyra@fias.uni-frankfurt.de

<sup>†</sup>Equal contributor: Marc Pereyra and Armin Drusko

<sup>1</sup> Frankfurt Institute for Advanced Studies (FIAS) and Goethe Universität Frankfurt am Main, Ruth-Moufang-Straße 1, 60438 Frankfurt am Main, Germany

Full list of author information is available at the end of the article

## Abstract

**Background:** The technical development of imaging techniques in life sciences has enabled the three-dimensional recording of living samples at increasing temporal resolutions. Dynamic 3D data sets of developing organisms allow for time-resolved quantitative analyses of morphogenetic changes in three dimensions, but require efficient and automatable analysis pipelines to tackle the resulting Terabytes of image data. Particle image velocimetry (PIV) is a robust and segmentation-free technique that is suitable for quantifying collective cellular migration on data sets with different labeling schemes. This paper presents the implementation of an efficient 3D PIV package using the Julia programming language—quickPIV. Our software is focused on optimizing CPU performance and ensuring the robustness of the PIV analyses on biological data.

**Results:** QuickPIV is three times faster than the Python implementation hosted in openPIV, both in 2D and 3D. Our software is also faster than the fastest 2D PIV package in openPIV, written in C++. The accuracy evaluation of our software on synthetic data agrees with the expected accuracies described in the literature. Additionally, by applying quickPIV to three data sets of the embryogenesis of *Tribolium castaneum*, we obtained vector fields that recapitulate the migration movements of gastrulation, both in nuclear and actin-labeled embryos. We show normalized squared error cross-correlation to be especially accurate in detecting translations in non-segmentable biological image data.

**Conclusions:** The presented software addresses the need for a fast and open-source 3D PIV package in biological research. Currently, quickPIV offers efficient 2D and 3D PIV analyses featuring zero-normalized and normalized squared error cross-correlations, sub-pixel/voxel approximation, and multi-pass. Post-processing options include filtering and averaging of the resulting vector fields, extraction of velocity, divergence and collectiveness maps, simulation of pseudo-trajectories, and unit conversion. In addition, our software includes functions to visualize the 3D vector fields in Paraview.

**Keywords:** Particle image velocimetry, Light-sheet fluorescence microscopy, Collective cell migration, Julia, 3D image analysis, *Tribolium castaneum*



## Background

Cellular migration in multi-cellular organisms often involves tissues or groups of cells that maintain stable or transient cell-cell contacts to preserve tissue integrity, sustain spatial patterning, or to enable the relocation of non-motile cells [1]. This phenomenon is generally known as collective cell migration, and it plays important roles in developmental processes, such as gastrulation or neural crest migration [2, 3], as well as in wound closure and cancer invasion [4]. Studies of collective cell migration on 2D cell cultures only partially reflect the physiology and architecture of *in vivo* tissues. Three-dimensional systems—such as model organisms, spheroids or organoids—are preferable, as they maintain physiological cell structures, neighborhood interactions, or mechanical extracellular properties, which have been recognized to play a role in regulating collective cellular migration [5–7]. Besides confocal fluorescence microscopy, light-sheet fluorescence microscopy (LSFM) has become one of the preferred techniques for three-dimensional imaging of biological samples, owing to its fast acquisition times, excellent signal-to-noise ratios, high spatial resolutions [8, 9], and low phototoxicity and photobleaching levels [10]. LSFM has been used to generate 3D time-lapse recordings of the complete embryonic morphogenesis of multiple model organisms [11, 12]. Based on light-sheet illumination, novel and improved imaging techniques are continuously being developed. For example, SCAPE (swept confocally-aligned planar excitation) microscopy offers more control over the viewing angle of the sample [13], while SVIM (selective volume illumination microscopy) dramatically increases acquisition times by dilating the light-sheet, at the expense of spatial resolution [14]. High temporal and spatial resolutions can be achieved with lattice light-sheet microscopy, where a combination of ultrathin light sheets and structured illumination are used [15]. The two last-mentioned techniques are particularly promising for resolving cellular migration and tissue rearrangements during quick morphogenetic events.

In order to quantify collective cellular migration in dynamic 3D biological data sets, we developed quickPIV, a free and open-source particle image velocimetry (PIV) package that offers fast and robust 3D, as well as 2D, PIV analyses. While several free and open-source 2D PIV software are readily available [16–19], the same is not true for 3D implementations. To the best of our knowledge, the Python implementation hosted in openPIV is the only other free and open-source PIV package that supports 3D analyses [18]. The fastest implementation in openPIV, however, corresponds to a 2D PIV implementation written in C++. Nevertheless, maintenance of this version was stopped in favor of the high-level and productive environment of its Python counterpart. In order to maximize performance without sacrificing productivity, our software is written in Julia, a modern programming language with high-level syntax similar to Python or Matlab that compiles to highly efficient code on par with C programs [20]. This choice is motivated by the high data volumes of 3D time-lapse recordings, which makes the analysis of multiple data sets computationally very expensive. For instance, a single sequence of 3D images of a developing embryo can easily reach data sizes of several Terabytes. Hence, the design principles of Julia enabled us to prioritize the CPU performance of quickPIV, and together with further optimizations, made it possible to reduce the processing speed of a pair of 3D volumes to several seconds.

The next subsection introduces PIV and discusses the strengths and limitations of applying PIV on biological samples. This is followed by a detailed description of the pipeline and the features implemented in quickPIV. The evaluation of our software includes a performance comparison to the C++ (2D) and Python (2D and 3D) implementations hosted in openPIV, as well as the accuracy evaluation of quickPIV on synthetic data. Furthermore, we analyze the ability of quickPIV to characterize migration patterns on three 3D time-lapse data sets of the embryonic development of the red flour beetle *Tribolium castaneum* [21, 22]. This is done by (1) simulating known translations on a 3D volume of *T. castaneum*, (2) validating the obtained vector fields against well-known migration patterns during the gastrulation of *T. castaneum*, and (3) by comparing the robustness of quickPIV on an embryo expressing both actin and nuclear molecular markers.

### Particle image velocimetry

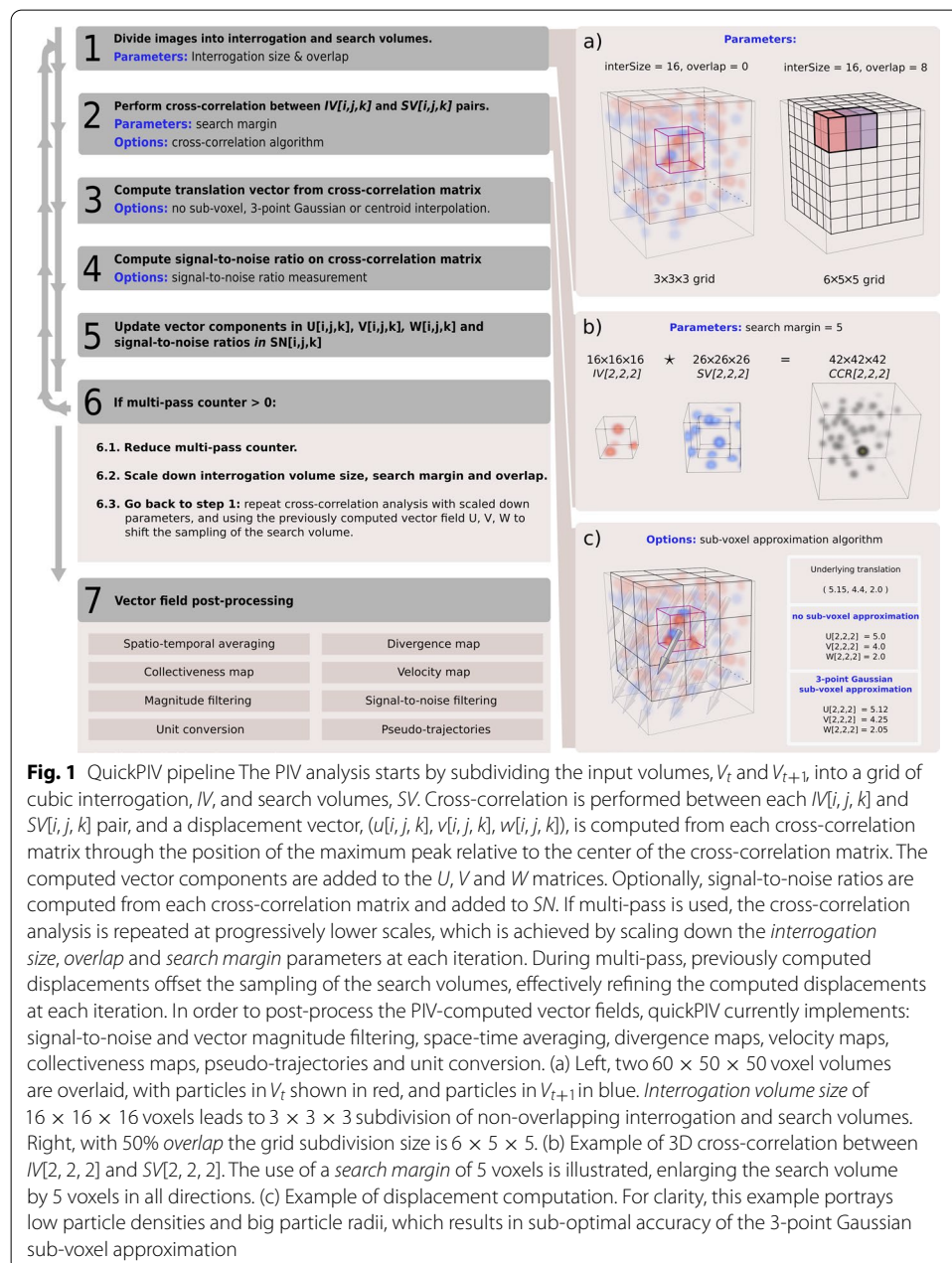
Particle image velocimetry is a segmentation-free technique developed and established in the field of fluid dynamics to obtain displacement fields describing the motion of small tracer particles suspended in a flowing medium [23]. If the density of seeding particles is not exceedingly high [24], the motion of each suspended particle can generally be recovered through particle tracking velocimetry (PTV) [25]. PTV is analog to single-cell tracking, requiring the segmentation of all particles in two consecutive recordings before establishing one-to-one correspondences between the particle positions. While the size and seeding density of the tracer particles in hydro- and aerodynamic PIV experiments can be tuned [26], the segmentability of biological samples is challenged by factors with no or limited experimental control. For example, cell segmentation is hindered by low contrast of the molecular marker, irregular cell morphologies, or high cell densities. Instead of detecting and tracking individual objects, PIV relies on cross-correlation to find the translation that best aligns the intensity patterns contained inside any given sub-region between two consecutive recordings. Vector fields are generated by extracting displacement vectors from multiple sub-regions across the input data [23].

The accuracy of PIV on biological data is mostly explained by the strengths and limitations of cross-correlation. In short, cross-correlation is a pattern-matching operation that is suitable for finding translations of the intensity distributions contained in two successive recordings [27]. Therefore, PIV is appropriate for quantifying collective cell migration, which is dominated by a common translation of the migrating group of cells. Moreover, the pattern-matching nature of cross-correlation extends the application of PIV to non-segmentable data sets, including unstained samples or those stained with any persistent intra-cellular marker. PIV has been used to quantify cell migration in 2D model systems, such as wound healing assays [28], tumor invasion [29, 30], skin patterning [31] and others [32–34]. Conversely, cross-correlation is challenged by transformations other than translations, such as rotations, shears or deformations. High temporal resolutions alleviate the contribution of these transformations by approximating them to local translations. Uncoordinated cellular migration also reduces the similarity of intensity patterns between successive recordings, which degrades the accuracy of PIV. However, if the cells are sufficiently different from each other such that they are unambiguously detected by cross-correlation, a PIV analysis matching the size of the cells can be used to effectively track the movement of independently migrating cell [35, 36].

### Implementation

This section outlines the three-dimensional PIV pipeline implemented in quickPIV. The workflow of a PIV analysis in quickPIV is illustrated in Fig. 1. This figure shows input volumes containing Gaussian particles to ease the visualization of the underlying translation. To accommodate all possible labeling schemes of biological samples, we generally refer to structures or intensity patterns in the analyzed data.

The input to a 3D PIV analysis is a pair of 3D volumes taken at consecutive time points,  $V_t[x, y, z]$  and  $V_{t+1}[x, y, z]$ , where  $(x, y, z)$  corresponds to the unique 3D coordinates of each voxel. Both input volumes are assumed to have the same dimensions. First,  $V_t$  is subdivided into a 3D grid of cubic sub-regions known as interrogation volumes,  $IV[i, j, k]$ , each



**Fig. 1** QuickPIV pipeline The PIV analysis starts by subdividing the input volumes,  $V_t$  and  $V_{t+1}$ , into a grid of cubic interrogation,  $IV$ , and search volumes,  $SV$ . Cross-correlation is performed between each  $IV[i, j, k]$  and  $SV[i, j, k]$  pair, and a displacement vector,  $(u[i, j, k], v[i, j, k], w[i, j, k])$ , is computed from each cross-correlation matrix through the position of the maximum peak relative to the center of the cross-correlation matrix. The computed vector components are added to the  $U$ ,  $V$  and  $W$  matrices. Optionally, signal-to-noise ratios are computed from each cross-correlation matrix and added to  $SN$ . If multi-pass is used, the cross-correlation analysis is repeated at progressively lower scales, which is achieved by scaling down the *interrogation size*, *overlap* and *search margin* parameters at each iteration. During multi-pass, previously computed displacements offset the sampling of the search volumes, effectively refining the computed displacements at each iteration. In order to post-process the PIV-computed vector fields, quickPIV currently implements: signal-to-noise and vector magnitude filtering, space-time averaging, divergence maps, velocity maps, collectiveness maps, pseudo-trajectories and unit conversion. (a) Left, two  $60 \times 50 \times 50$  voxel volumes are overlaid, with particles in  $V_t$  shown in red, and particles in  $V_{t+1}$  in blue. *Interrogation volume size* of  $16 \times 16 \times 16$  voxels leads to  $3 \times 3 \times 3$  subdivision of non-overlapping interrogation and search volumes. Right, with 50% *overlap* the grid subdivision size is  $6 \times 5 \times 5$ . (b) Example of 3D cross-correlation between  $IV[2, 2, 2]$  and  $SV[2, 2, 2]$ . The use of a *search margin* of 5 voxels is illustrated, enlarging the search volume by 5 voxels in all directions. (c) Example of displacement computation. For clarity, this example portrays low particle densities and big particle radii, which results in sub-optimal accuracy of the 3-point Gaussian sub-voxel approximation

specified by its position in the grid,  $(i, j, k)$ . The dimensions of the grid subdivision are determined by the interrogation volume size and the overlap between adjacent interrogation volumes, see Fig. 1a. For each interrogation volume, a corresponding search volume,  $SV[i, j, k]$ , can be defined in  $V_{t+1}$ .

Structures moving inside  $IV[i, j, k]$  by a translation  $\mathbf{s} = (s_x, s_y, s_z)$  are expected to be found  $\|\mathbf{s}\|$  voxels away in the direction of the translation in  $SV[i, j, k]$ . The underlying translation,  $\mathbf{s}$ , of the structures contained in  $IV[i, j, k]$  and  $SV[i, j, k]$  is recovered through a cross-correlation analysis [27]. The cross-correlation between a pair of interrogation and search volumes results in a 3D cross-correlation matrix. In the absence of other transformations, the vector from the center to the maximum peak of the cross-correlation matrix reflects the underlying translation of the structures contained in  $IV[i, j, k]$  and  $SV[i, j, k]$ . The structures visible in  $IV[i, j, k]$  may move outside the borders of the corresponding  $SV[i, j, k]$ . This is known as out-of-frame loss, and it limits the ability of cross-correlation to match the spatial intensity distributions between the pair of interrogation and search volumes. This can be compensated by enlarging the search volumes by a given margin along all dimensions, designated as search margin in quickPIV. The search margin should not be much larger than the expected translation strength of the structures, as enlarging the search volumes comes at the expense of performance. Figure 1b depicts the cross-correlation of the central interrogation and search volumes in Fig. 1a, including a search margin of 5 voxels around the search volume.

By computing a displacement vector for each pair of interrogation and search volumes, PIV analyses generate a vector field that describes the velocity distribution of the structures contained in the input volumes. The components of the PIV-computed vector field are returned separately in three 3D matrices:  $U$ ,  $V$  and  $W$ . It should be noted that the resolution of the final vector field is decided by the size of the interrogation volumes and their overlap, which determine the grid subdivision of  $V_t$  and  $V_{t+1}$ . Multi-pass is implemented to overcome this trade-off between resolution and the interrogation size of the PIV analysis.

### Cross-correlation

The cross-correlation of two one-dimensional real-valued functions is defined as:

$$[f \star g](s) = \int_{-\infty}^{\infty} f(x)g(x+s)dx, \quad (1)$$

where  $s$  has the effect of shifting  $g(x)$  along the  $x$ -axis. Cross-correlation involves computing the dot product of  $f(x)$  and  $g(x+s)$  for all possible values of  $s$ . Since the dot product entails a basic measure of similarity, the value of  $s$  that achieves the highest dot product represents the translation that best aligns the two functions.

The form of cross-correlation in Eq. (1) is known as spatial cross-correlation. Discrete implementations of spatial cross-correlation have a 1D complexity of  $O(N^2)$ . Taking advantage of the convolution theorem, cross-correlation can be computed in the frequency domain through Fourier transforms of  $f(x)$  and  $g(x)$ :

$$f \star g = \mathcal{F}^{-1}\{\overline{\mathcal{F}\{f\}} \cdot \mathcal{F}\{g\}\}, \quad (2)$$

where  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  denote the Fourier and inverse Fourier transforms, respectively. Each Fourier and inverse Fourier transform in Eq. (2) can be computed efficiently with the

Fast Fourier Transform (FFT) algorithm [37], which has a 1D complexity of  $O(N \log N)$ . Since Eq. (2) does not involve any operations with higher complexities than FFT's, the overall complexity of 1D cross-correlation in the frequency domain is  $O(N \log N)$ . For this reason, cross-correlation in quickPIV is computed in the frequency domain. We rely on a Julia wrapper around the mature and optimized Fastest Fourier Transform of the West (FFTW) C library [38] to compute all Fourier and inverse Fourier transforms. FFTW implementations of FFT generalize to multi-dimensional data, enabling the efficient three-dimensional computation of cross-correlation.

To tackle the bias of the dot product towards high intensities, we implemented zero-normalized cross-correlation (ZNCC). Considering  $IV$  and  $SV$  as a pair of 3D interrogation and search volumes, ZNCC is calculated at each translation of  $IV$  by:

$$ZNCC[\mathbf{s}] = \sum_{x,y,z} \frac{(IV[\mathbf{x}] - \mu_{IV})(SV[\mathbf{x} + \mathbf{s}] - \mu_{SV})}{\sqrt{\sum_{x,y,z} (IV[\mathbf{x}] - \mu_{IV})^2 \sum_{x,y,z} (SV[\mathbf{x} + \mathbf{s}] - \mu_{SV})^2}} \quad (3)$$

where  $\mathbf{x}$  is a 3D index  $(x, y, z)$  running over all voxels of  $IV$ ,  $\mathbf{s}$  is the displacement vector  $(s_x, s_y, s_z)$ , and  $\mu_{IV}$  and  $\mu_{SV}$  are the average intensity values of the interrogation and search volumes, respectively. Zero-normalized cross-correlation is implemented efficiently in quickPIV following the work of Lewis, who noted that the numerator in Eq. (3) can be computed efficiently in the frequency domain, while each sum in the denominator can be calculated with eight operations from an integral array of the search volume [39].

To further improve the pattern-matching robustness of cross-correlation, quickPIV also offers normalized squared error cross-correlation (NSQECC). At each translation of  $IV$ , NSQECC is computed as [40]:

$$NSQECC[\mathbf{s}] = \sum_{x,y,z} \frac{(IV[\mathbf{x}] - SV[\mathbf{x} + \mathbf{s}])^2}{\sqrt{\sum_{x,y,z} (IV[\mathbf{x}])^2 \sum_{x,y,z} (SV[\mathbf{x} + \mathbf{s}])^2}}, \quad (4)$$

where  $\mathbf{x}$  is a 3D index  $(x, y, z)$  running over all voxels of  $IV$ , and  $\mathbf{s}$  is the displacement vector  $(s_x, s_y, s_z)$ . Following the example of [39], Eq. (4) is implemented efficiently in quickPIV by expressing the numerator and denominator in terms of three components:  $\sum (IV[\mathbf{x}])^2$ , which is constant,  $\sum (SV[\mathbf{x} + \mathbf{s}])^2$ , which is computed efficiently for each translation from an integral array, and  $-2 \sum (IV[\mathbf{x}]SV[\mathbf{x} + \mathbf{s}])$ , which can be computed as an unnormalized cross-correlation in the frequency domain. For convenience, quickPIV implements the inverse of Eq. (4),  $1/(1 + NSQECC[\mathbf{s}])$ , to obtain a maximum peak at the translation that minimizes the differences between the interrogation and search volumes.

### Peak sub-voxel approximation

In order to detect non-integer translations, two sub-voxel interpolation methods are included in quickPIV: the centroid-based and the 3-point Gaussian sub-voxel approximations [41]. In both methods, sub-voxel refinements are computed by considering the direct neighboring values around the maximum peak of the cross-correlation matrix. The centroid-based sub-voxel refinements,  $\Delta$ , are computed by



$$\Delta[\mathbf{d}] = \frac{C[\mathbf{x} + \mathbf{d}] - C[\mathbf{x} - \mathbf{d}]}{C[\mathbf{x} + \mathbf{d}] + C[\mathbf{x}] + C[\mathbf{x} - \mathbf{d}]}, \quad (5)$$

where  $C$  refers to the cross-correlation matrix,  $\mathbf{x}$  are the voxel coordinates of the maximum peak in the cross-correlation matrix, and  $\mathbf{d}$  is the standard basis vector for each dimension, e.g. (1, 0, 0) for the first dimension. Following the same notation, the 3-point Gaussian sub-voxel refinement of the integer displacement is given by

$$\Delta[\mathbf{d}] = \frac{\ln(C[\mathbf{x} + \mathbf{d}]) - \ln(C[\mathbf{x} - \mathbf{d}])}{2 \ln(C[\mathbf{x} + \mathbf{d}]) - 4 \ln(C[\mathbf{x}]) + 2 \ln(C[\mathbf{x} - \mathbf{d}])}. \quad (6)$$

To acquire sub-voxel precision, the interpolated  $\Delta$  is added to the integer displacement vector from the maximum peak to the center of the cross-correlation matrix. QuickPIV defaults to the 3-point Gaussian sub-voxel approximation, which performs particularly well when the input volumes contain Gaussian particles, as the convolution of Gaussians produces another Gaussian distribution [42].

### Multi-pass

We implemented a multi-pass procedure to increase the accuracy of the PIV analysis and to extend its dynamic range, i.e., the range of detectable displacements. While a search margin can be added to increase the dynamic range of a standard PIV analysis, it does not eliminate the dependence on small interrogation volumes to achieve high resolutions, which limits the specificity and enhances the noise of the intensity patterns contained in the interrogation volumes [43]. Alternatively, high resolutions with good dynamic ranges can be achieved by combining large interrogation volumes with high overlaps. However, this approach is computationally expensive and increases the final resolution by adding redundancy between consecutive cross-correlation computations [44].

The multi-pass algorithm starts the PIV analysis with up-scaled interrogation and search volumes, followed by iterative rounds of PIV analyses with gradually smaller interrogation size and search volumes. Additionally, the displacements calculated during previous rounds are used to offset the sampling of the search volumes at future rounds [45]. The multi-pass factor  $f$  defines the number of total rounds that will be conducted. Therefore, multi-pass is enabled by setting  $f$  larger than 1. At each multi-pass round, the interrogation size, search margin and overlap parameters are scaled with respect to their user-defined values. The value of these parameters in each round  $r$  is computed as follows:

$$\kappa_r = (1 + f - r) * \kappa_0, \quad (7)$$

where  $\kappa_0$  designates the user-defined value for interrogation size, search margin or overlap,  $\kappa_r$  is the up-scaled value of these parameters at round  $r$ , and  $f$  is the multi-pass factor. The final round is performed with a factor of 1, i.e., the initial interrogation sizes.

### Post-processing

Some of the post-processing features explained below include local information around the vector being processed. In such cases, a square (2D) or cubic

(3D) region is sampled around each post-processed vector. For instance,  $r_x$  and  $r_y$  define a square area around an arbitrary vector in a 2D vector field,  $v_{ij}$ , given by  $L = \{v_{i+r_x, j+r_y} \mid -r \leq r_x \leq r \text{ and } -r \leq r_y \leq r\}$ .

### Filtering

A PIV-computed vector is considered unreliable if it was computed from a cross-correlation matrix containing multiple peaks with similar heights as the maximum peak. This reveals uncertainty about the underlying displacement, which might be caused by unspecific structures, background noise and/or loss of structure pairs [46, 47]. QuickPIV adopts the primary peak ratio,  $PPR$ , to measure the specificity of each computed vector,

$$PPR = \frac{C_{\max 1}}{C_{\max 2}}, \quad (8)$$

where  $C_{\max 1}$  is the height of the primary peak in the cross-correlation matrix and  $C_{\max 2}$  is the height of the secondary peak. Vectors with high  $PPR$  values are considered to have high signal-to-noise ratios [48]. Therefore, quickPIV offers filtering of unreliable vectors by discarding those vectors with a  $PPR$  value lower than a given threshold,  $th_{PPR}$  [48].

Additionally, quickPIV includes both global and local filtering in terms of vector magnitudes. Currently, quickPIV offers low pass and high pass filters of vector magnitudes, which can be concatenated to perform band-pass filtering. Global magnitude filtering can also be performed on those vectors whose magnitude is more than a certain number of standard deviations away from the mean magnitude of the vector field. Local magnitude filtering is implemented by discarding vectors whose magnitude is at least  $n$  standard deviations away from the mean magnitude, computed in a radius  $r$  around each vector.

All filtering functions in quickPIV accept an optional argument that is used to determine the replacement scheme of the filtered vectors. Currently, quickPIV offers three replacement functions: zero-replacement, mean replacement and median replacement. The former sets all components of the filtered vectors to zero. Both the mean and median replacement schemes are parametrized by the radius of the neighboring region used to compute the mean or median vector.

### Spatial and temporal averaging

Spatial and spatio-temporal averaging of the computed vector fields are included in quickPIV. Spatial averaging depends on one parameter: the radius,  $r_s$ , of the considered neighboring region around each vector. Different radii for each dimension can be provided by passing an array of values,  $[r_x, r_y, r_z]$ . Spatio-temporal averaging considers two parameters: the averaging radius in space and the number,  $n_t$ , of adjacent vectors along the time axis considered in the temporal averaging, e.g.  $\{v_{i,j,k,t+r} \mid -n_t \leq r \leq n_t\}$ .

### Similarity-selective spatial averaging

Spatial averaging tends to dissolve vectors adjacent to the background and creates artificial vectors in regions containing dissimilar vectors. A similarity-selective spatial averaging has been developed to overcome these limitations, and to enhance the visualization of collective migration. Two vectors are considered to be similar if they point in



the same direction, which is established if their normalized dot product is greater than a user-defined threshold. Given any vector in the PIV-computed vector field,  $\mathbf{v}[i, j, k]$ , an average vector is built by considering only those neighboring vectors at a radius  $r$  that are similar to  $\mathbf{v}[i, j, k]$ . The averaged vector is then normalized to unit length, and its magnitude is further re-scaled by the ratio between the number of similar neighboring vectors and the total number of neighboring vectors. Therefore, the effect of similarity-selective averaging is to average the direction of each vector among similar neighboring vectors, and to re-scale the magnitude of each vector by the local collectiveness.

### ***Mappings***

QuickPIV provides functions for extracting several relevant quantities from the PIV-computed vector fields. Velocity maps are generated by returning the magnitude of each vector from a given vector field. QuickPIV implements convergence/divergence mappings to detect the presence of sinks and sources in the PIV-computed vector fields. This is done by generating a cube of normalized vectors that either converge (sink) or diverge (source) from the center of the cube, and cross-correlating this cube with the normalized vector field. This mapping is parametrized by the size of the cube, which determines the scale of the convergence/divergence map. Collectiveness maps are built by computing the number of neighboring vectors at a radius  $r$  from each vector in the vector field  $v_{i,j}$  whose normalized dot product is greater than a threshold.

### ***Pseudo-trajectories***

Pseudo-trajectories can be generated with quickPIV to visualize the approximate paths of cells and tissues from the PIV-computed vector fields. When computing pseudo-trajectories, a user-defined number of particles is randomly distributed within the dimensions of the vector field. The position of each particle is rounded to integer coordinates in order to sample a displacement from the vector field, which shifts the particle from its current position. By repeating this process, a three-dimensional path is obtained for each simulated particle. It is possible to constrain the computation of pseudo-trajectories to a period of interest by specifying the start and end time points. Moreover, spatially interesting regions can be selected by specifying the spatial range over which to initialize the positions of the particles.

### ***Conversion to physical units***

Last but not least, to convert voxel displacements into physically meaningful velocities both the frame rate and the physical units of each voxel dimension need to be taken into account. These values can be provided during the creation of the PIV-parameter object and quickPIV will automatically re-scale the resulting vector field after the analysis.

### **QuickPIV accuracy evaluation**

The correct implementation of a PIV analysis depends on its ability to detect translations. Accordingly, the accuracy of quickPIV is assessed by generating pairs of artificial images and volumes containing synthetic particles related by a known translation. Synthetic particles are rendered according to [49]. The bias and random errors are computed to evaluate the agreement of quickPIV predictions to the known translations [49]:

$$\epsilon_{\text{bias}} = \frac{1}{n} \sum_{i=1}^n |d_{\text{PIV},i} - d_{\text{true}}| \quad (9)$$

$$\epsilon_{\text{rand}} = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_{\text{PIV},i} - \overline{d_{\text{PIV}}})^2} \quad (10)$$

where  $d_{\text{PIV},i}$  is the  $i^{\text{th}}$  PIV-computed displacement,  $d_{\text{true}}$  is the known translation,  $\overline{d_{\text{PIV}}}$  is the average PIV-computed displacement and  $n$  is the number of repeats. The bias and random errors represent the accuracy and the precision of quickPIV's approximation of the underlying translation, respectively. The effect of the following parameters on the accuracy of quickPIV are evaluated, both in 2D and 3D: interrogation size, particle density, particle diameter, 3-point Gaussian sub-pixel approximation and the use of a search margin to correct for out-of-frame loss.

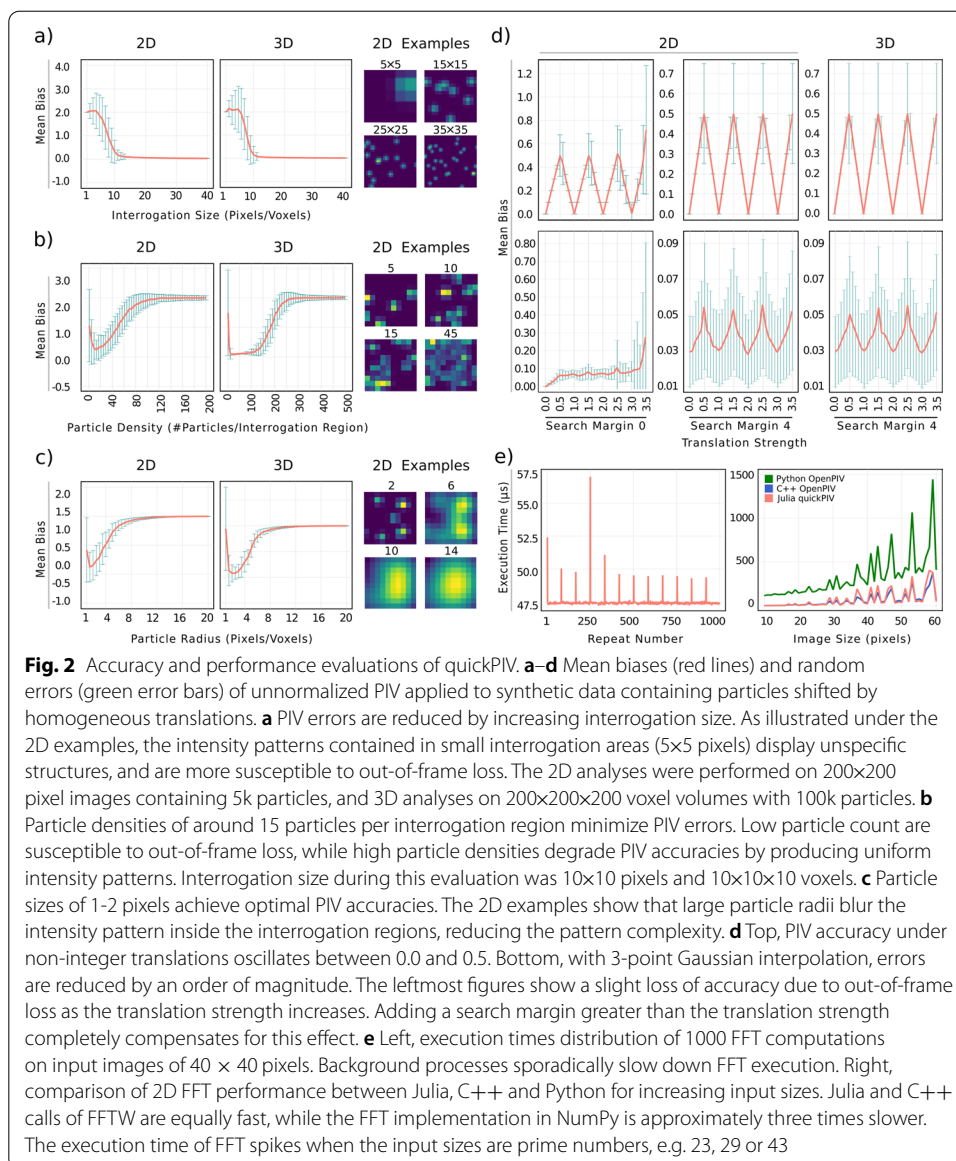
#### QuickPIV performance evaluation

The performance of our software is evaluated by comparing the execution times of quickPIV with those of the C++ and Python implementations hosted in openPIV. First, we analyzed the time required to compute cross-correlation in the frequency domain with the three packages. By comparing the execution times of quickPIV and the C++ implementation, we can determine whether calling the FFTW C-library from Julia adds any noticeable overhead compared to C++. Since the Python implementation uses the NumPy library to compute the Fourier and inverse Fourier transforms, this test also reveals any performance differences between FFTW and NumPy. On the other hand, we compare the execution time of complete 2D and 3D PIV analyses between the three PIV packages. The set of parameters used in these PIV analyses are listed in the description of Table 1.

For the sake of using a common benchmarking pipeline, language-specific packages for measuring the execution times are avoided. Each execution time measurement shown in Fig. 2e corresponds to the minimum execution time from 1000 repeated measurements. Taking the minimum execution time filters out random delays originating from background processes [50]. The left panel in Fig. 2e illustrates the interference of background processes in the distribution of 1000 execution measurements of FFT cross-correlation. All measurements presented below were performed on a machine with an Intel Core i5-8300H processor  $4 \times 2.3$  GHz. All PIV analyses were executed on a single thread.

#### QuickPIV on the embryogenesis of *Tribolium castaneum*

To test the accuracy of quickPIV on biological data, we analyzed three 3D time-lapse data sets of the embryonic development of *T. castaneum*: (1) two embryos from a hemizygous transgenic line that ubiquitously expresses nuclear-localized mEmerald and (2) one embryo from a double hemizygous transgenic line that expresses nuclear-localized mRuby2 ubiquitously and actin-binding Lifeact-mEmerald only in the serosa [22]. Using LSM, the embryos were recorded at intervals of (1) 30 minutes or (2) 20 minutes along 4 directions in rotation steps of  $90^\circ$  around the anterior-posterior axis in (1) one or (2) two fluorescence channels [21]. The four directions



were fused according to Preibisch et al. [51] to generate evenly illuminated volumes with isotropic resolution. The fused volumes were cropped to 1000 × 600 × 600 voxels (height,width,depth), the embryos were manually placed in the center of the volumes and their anterior–posterior axis was manually aligned with the vertical axis.

Three time points during gastrulation were analyzed with quickPIV in the two embryos of data set (i). Two time points of the double hemizygous transgenic line (ii) were analyzed in both channels, allowing to compare the vector fields obtained from the Lifeact-mEmerald actin signal with those from the nuclear-localized mRuby2 marker. The PIV analyses were performed on both data sets with NSQECC. The vector fields resulting from these analyses are shown in Figs. 3 and 4, post-processed with similarity-selective averaging with an averaging radius of 2 neighboring vectors

and a similarity threshold of 0.5. The visualization of the embryo volumes and the computed vector fields has been done in Paraview 5.7.0.

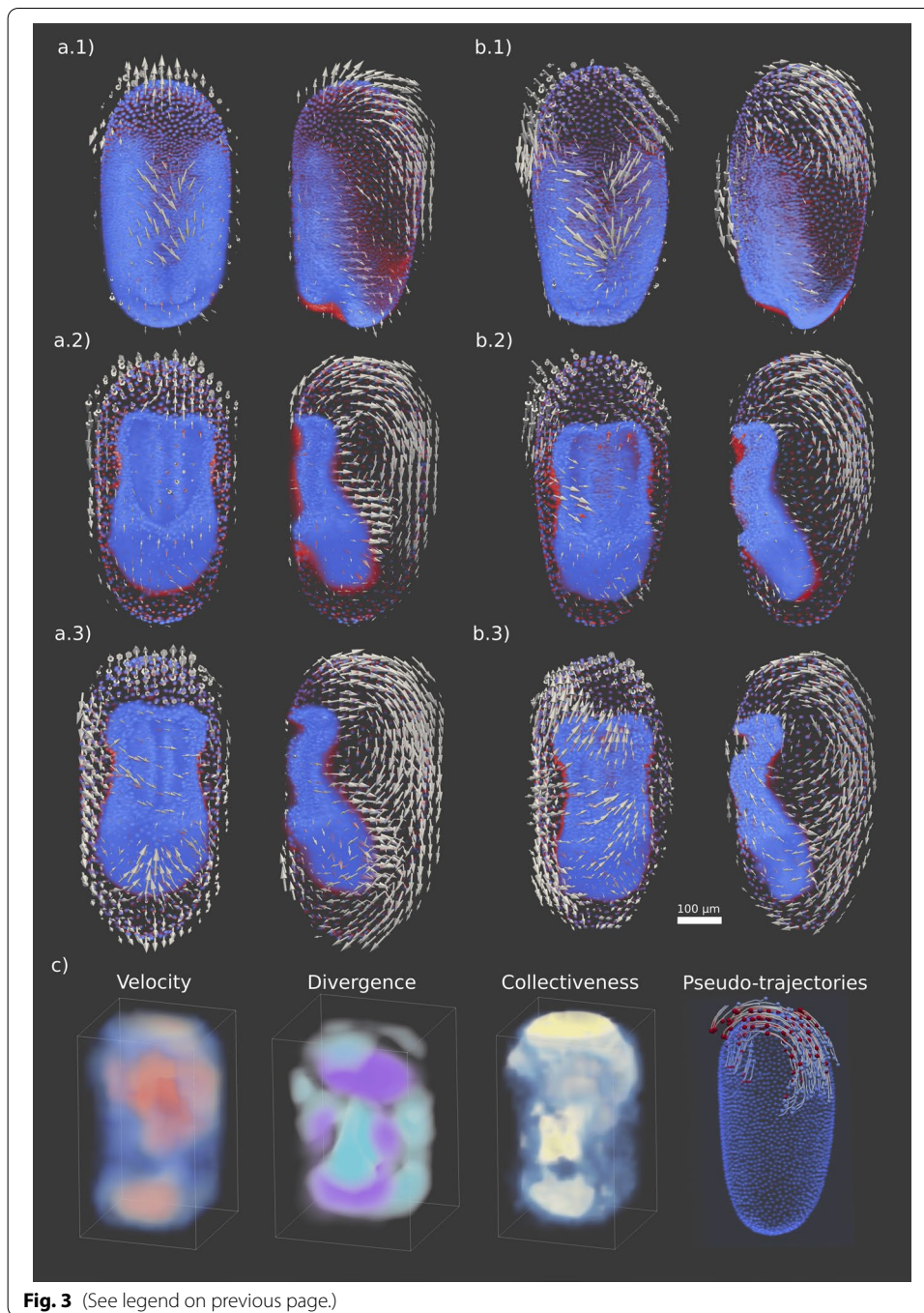
## Results

The accuracy evaluation of quickPIV quantitatively reproduces the expected accuracies described in the PIV literature, attesting the correctness of our PIV implementation [52–55]. Our analysis shows a monotonic decrease of the total error (bias and random errors) with increasing interrogation sizes [55], reaching errors as low as  $0.02 \pm 0.01$  pixels/voxels (Fig. 2a). This is the expected behavior in our synthetic tests, since all simulated particles are subjected to the same translation. Our results also agree on the presence of optimal values for both particle density and particle size [52]. It can be appreciated from the 2D examples included in Fig. 2b and c that high particle densities and large particle sizes generate diffuse images that can not be unambiguously matched by cross-correlation. Without sub-pixel/voxel interpolation, the PIV analysis cannot capture the decimal components of the simulated translations, shown in the top row of Fig. 2d [52, 53]. As described in the literature, the 3-point Gaussian sub-pixel approximation reduces this error by one order of magnitude (bottom row in Fig. 2d) [56]. Moreover, search margins are needed to counteract the out-of-frame errors induced by increasing translation (Fig. 2d, left panel). A search margin of 4 pixels/voxels (Fig. 2d, middle and right panels) completely compensates this effect for all simulated translations.

We performed an analogous accuracy analysis on the *T. castaneum* data set, where we quantified the accuracy of quickPIV in detecting known translations on one 3D volume in data set (i). We observed that diffuse and unspecific patterns in the embryo induce biases when using ZNCC. These biases are completely avoided by using NSQECC, which detects the underlying translation with 100% accuracy given a sufficiently large search margin (see Figure S1). We further analyzed the height distribution of the maximum cross-correlation peaks during the PIV analysis with NSQECC of two consecutive volumes of *T. castaneum*, shown in Fig. 4e. High peaks are found in the collectively migrating serosa cells at the anterior pole of the embryo, which we classify as segmentable and trackable (S/T), and in the non-segmentable and non-trackable (NS/NT) gastrulating embryo, Fig. 4d. These high peaks indicate that cellular migration in these regions is well approximated by a collective translation, and

(See figure on next page.)

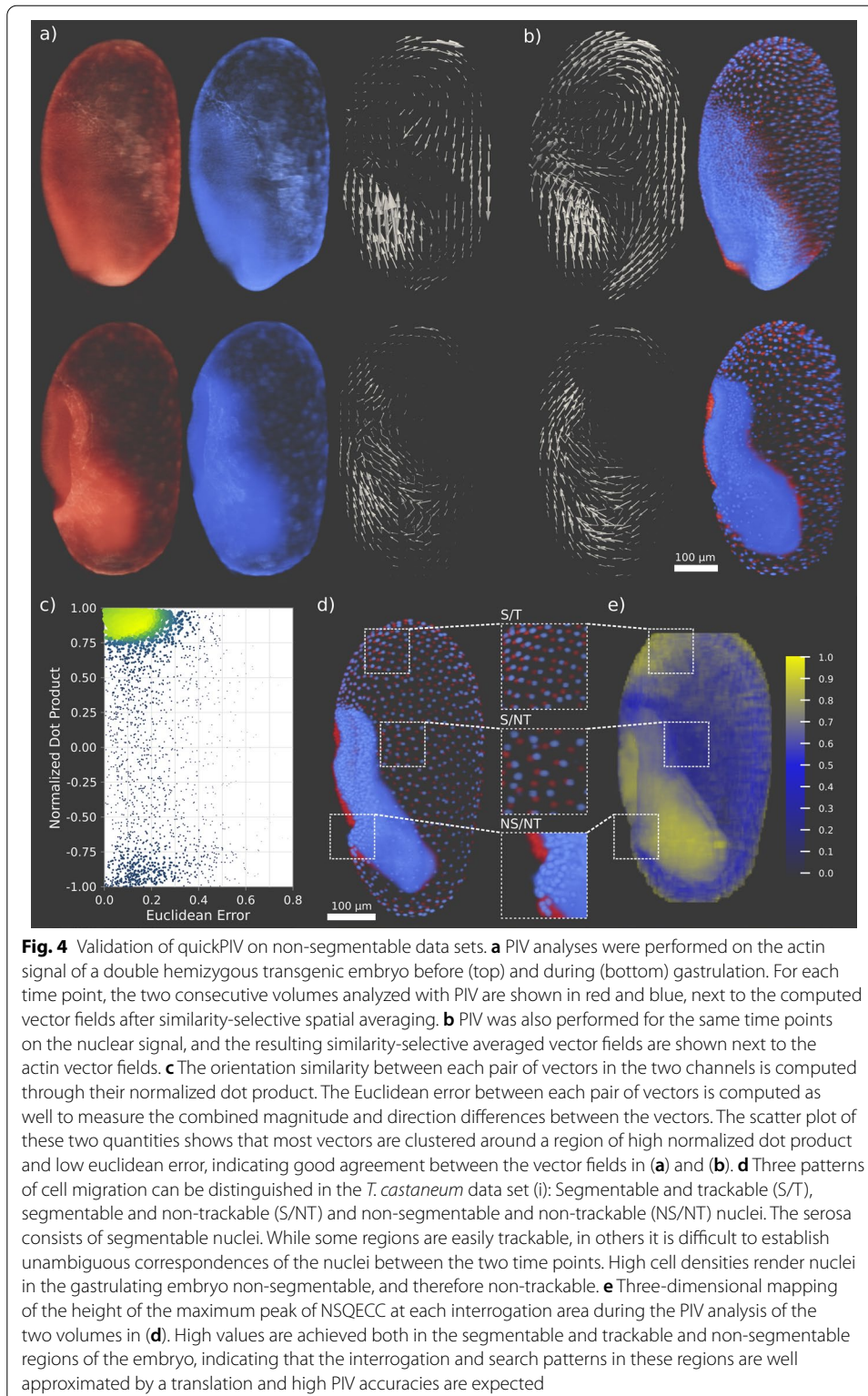
**Fig. 3** 3D PIV analysis on the embryogenesis of two *T. castaneum* embryos. Each vector field in **a**. 1–3 and **b**. 1–3 is plotted on top of the two volumes it was computed from, where the red signal corresponds to the initial time point and blue intensities belong to the consecutive time point. A few spurious vectors obtained on the background due to the fluorescence bleeding from the embryo were manually curated. Embryos are shown from their ventral and lateral sides. **a-b.1** At the onset of gastrulation, serosa nuclei at the anterior end of both analyzed embryos collectively spread towards the dorsal side of the embryos. Moreover, the central and posterior regions on the ventral side undergo coordinated condensation movements that will later give rise to the internalizing germband. **a-b.2** The wide-spread serosa cells over the anterior pole and the dorsal side engage in a highly coordinated movement of the tissue towards the posterior pole. Time points **a-b.3** are characterized by a highly collective flow of serosa cells towards the ventral side, leading to the emergence and closing of the serosa window. Serosa cells at the anterior pole, dorsal side and the posterior pole collectively migrate clock-wise towards the ventral midline, giving rise to a cell migration pattern resembling a vortex. **c** Exemplary post-processing analyses applied to the vector field shown in a.1. From left to right: velocity map showing higher velocities in red, divergence(purple)/convergence(cyan) map, collectiveness map displaying higher local collectiveness in yellow, and pseudo-trajectories at the anterior pole of the embryo in a.1) over 10 time points (5 h)



**Fig. 3** (See legend on previous page.)

that the intensity patterns between the interrogation and search volumes are not deformed, rotated or sheared significantly. Non-collective migration of the serosa cells reduces the height of the NSQECC peaks in the central regions of the extraembryonic membranes, which we consider to be segmentable but not easily trackable (*S/NT*), since cell correspondences between the two time points can not unambiguously be assessed visually, Fig. 4d. A three-dimensional visualization of the maximum peak distribution in Fig. 4e is provided in Video S1.





Performance-wise, calling the FFTW C-library is equally efficient from Julia and C++ (see Fig. 2e, right). In contrast, the performance of the NumPy implementation of the FFT algorithm is three times slower than the one provided in the FFTW library.



This performance difference is translated to the complete 2D and 3D PIV analyses of the PIV packages, where the Python implementation in openPIV is consistently three times slower than both the C++ implementation (2D) and quickPIV (2D and 3D), see Table 1. Our results also show that 2D PIV analyses are performed faster with quickPIV than with the C++ implementation in openPIV. Since both packages share the same cross-correlation performance, this difference can only be explained by compiler optimizations brought by Julia's compilation pipeline, or by the ease of implementing good programming practices in Julia's high-level environment. For instance, quickPIV avoids bound checks when possible, minimizes memory allocations by using in-place operations, and leverages SIMD (single instruction, multiple data) operations exposed by the Julia programming language.

From a practical standpoint, we found that performance of PIV analyses can be dramatically increased by subsampling the input volumes and removing the background interrogation areas from the PIV analysis. For example, a PIV analysis of two volumes of *T. castaneum* with the following parameters (interSize of 60 voxels, searchMargin of 0 voxels, overlap of 30 voxels and multi-pass factor of 2), while skipping interrogation volumes with a maximum intensity lower than 100, takes 29 minutes. After applying subsampling by a factor of 3, the analogous analysis on the subsampled data (interSize of 20 voxels, searchMargin of 0 voxels, overlap of 10 voxels and multi-pass factor of 2) takes 55 s to complete. The results shown in Figs. 3 and 4, which were obtained after subsampling the input volumes by a factor of three in all dimensions, are in full agreement with the same analyses performed without subsampling (Fig. S2). The spatial resolution in this data set was very high, which is necessary to discern smaller structures. For motion analysis a lower image resolution is sufficient to obtain the same results. Before subsampling images, we, however, advise to test the agreement between the PIV vector fields in the original and a subsampled image.

The application of quickPIV to the two *T. castaneum* embryos of data set (i) is shown in Fig. 3. The red and blue intensities correspond to the nuclear signal of the first and second input volumes, respectively, which aids in visualizing the underlying displacement of the nuclei between each pair of analyzed time points. The vector fields at the anterior regions of the embryos in Figs. 3a.1 and b.1 capture the underlying radially diverging pattern of cell migration towards the dorsal side of the embryo. Our PIV analyses also capture the coordinated condensation movement of the cells in the central and posterior regions, which will later give rise to the germband. These regions exhibit high

**Table 1** Performance evaluation of complete PIV analyses

	C++	Python	quickPIV
2D	63 ms	160.81 ms	50.42 ms
3D	-	59.72 s	18.09 s

2D analyses were performed on a pair of images of size 512×512 pixel, with the following PIV parameters: FFT cross-correlation, interrogation size of 32 pixels, no search margin, overlap of 16 pixels, no multi-pass, 3-point Gaussian subpixel approximation and peak-to-peak signal-to-noise algorithm. 3D analyses were performed on volumes with dimensions 512×512×123 pixels, and the following PIV parameters: FFT cross-correlation, interrogation size of (49, 49, 11), no search margin, overlap of (14, 14, 3), no multi-pass, 3-point Gaussian sub-voxel approximation and peak-to-peak signal-to-noise algorithm. The reported measurement are the minimum execution time from 1000 and 100 repeats, for the 2D and 3D analysis respectively

cellular densities, challenging visual examination and rendering nuclei segmentation and tracking approaches unfeasible. Figure 3a.2 and b.2 are characterized by vastly coordinated movements of the wide-spread serosa cells over the anterior pole and along the dorsal side towards the posterior pole. Figures 3a.3 and b.3 depict a highly coordinated flow of serosa cells from the dorsal side over both, the posterior pole and the lateral equator, towards the ventral side, where they eventually give rise to the serosa window [57]. These observations are not only consistent with previous studies of collective cell migration during the gastrulation of *T. castaneum*, which were obtained through 1D PIV analyses [58] and manual 2D tracking of the extra-embryonic serosa cells [59, 60], but for the first time describe this process in 3D. Figure 3c illustrates the velocity, divergence/convergence and collectiveness mappings as well as some computed pseudo-trajectories on the anterior region of the embryo.

Finally, the results from the analysis of the double hemizygous transgenic line, (ii), demonstrate the robustness of quickPIV on non-segmentable data. The agreement of the vector fields on the anterior pole of the embryo (which is non-segmentable in the actin signal, segmentable in the nuclear signal and exhibits high degrees of collective cell migration in both channels) indicates that PIV accuracies are independent of the segmentability of the input data sets, Figs. 4a and b. A quantitative comparison of the PIV vector fields between the nuclear and the actin stained volumes shows a high degree of similarity. This is illustrated in the scatter plot shown in Fig. 4c, exhibiting a high density in the area of large dot products and small Euclidean errors. The similarity of the actin and nuclear vector fields in highly dense non-segmentable regions further underlines the robustness of quickPIV regardless of the labeling scheme of the data sets.

## Conclusions

QuickPIV represents a free and open-source solution for performing efficient and robust quantification of collective cellular migration in the increasingly popular 3D dynamic data sets in life sciences. Our software includes several well established PIV features, such as multi-pass and sub-voxel peak approximation, as well as post-processing functions and visualization of the 3D vector fields in Paraview.

To our knowledge, quickPIV is the only free PIV software that offers normalized squared error cross-correlation (NSQECC), which we found to be necessary for accurately describing collective cell and tissue migration in non-segmentable data sets. By using NSQECC, we could quantify collective cell migration from the non-segmentable and highly dynamic actin signal in a double hemizygous transgenic embryo of *T. castaneum*. The resulting vector fields were in complete agreement with previously published descriptions of the gastrulation movements in *T. castaneum*, and showed a strong correlation with the vector fields obtained from the nuclear signal of the same embryo. Moreover, the height distribution of the maximum cross-correlation peaks further indicates that NSQECC is robust to non-segmentable data.

The performance evaluation of quickPIV shows that our software is three times faster in 2D and 3D analyses than the Python PIV implementation in openPIV, and also faster than the 2D implementation written in C++. This performance advantage is only possible because of the design of the Julia programming language and the optimization possibilities that it provides. By considering subsampling and excluding unnecessary regions

of the input data (such as empty background), the quickPIV analysis of a pair of 3D volumes can be reduced to several seconds. These speeds are compatible with real-time PIV analyses, enabling the integration of PIV pipelines into smart microscopy techniques. For example, vector fields obtained with quickPIV could be used to automatically detect the onset of developmental events and adjust the acquisition parameters accordingly, e.g. laser power or acquisition interval.

Overall, we believe that 3D PIV analyses will play an important role in understanding 3D biological processes as novel 3D imaging techniques are developed and adopted. For example, SVIM can already achieve up to 100× higher recording speeds than standard LSFM. Such high temporal resolutions increase the accuracy of PIV and make PIV the ideal solution for reliable and automated pipelines for quantifying collective cellular migration. However, the computational demands required to analyze such temporally resolved data sets can only be met by further optimizations of quickPIV's performance. Therefore, future efforts will be directed towards adding multi-threading support and implementing our PIV analyses on a graphics card [61].

### Availability and requirements

Project name: quickPIV

Project home page: <https://github.com/Marc-3d/quickPIV>

Operating system(s): Platform independent

Programming language: Julia

Other requirements: Julia1.3.1 or higher

License: MIT License

Any restrictions to use by non-academics: None.

### Supplementary Information

The online version contains supplementary material available at <https://doi.org/10.1186/s12859-021-04474-0>.

**Additional file 1.** This file contains a video animating the rotation and slicing of the three-dimensional distribution of the maximum peak heights at each interrogation area during the PIV analysis using NSQECC of two 3D volumes of data set (i).

**Additional file 2.** This file contains **Figs. 1** and **2**. **Figure S1** reports the results from our synthetic accuracy evaluation on biological data, including the comparison between the accuracies obtained with ZNCC and NSQECC PIV analyses. **Figure S2** shows the same PIV analyses as Fig. 3, with the difference that no subsampling of the input volume was used to obtain the results in Figure S2.

### Acknowledgements

Not applicable

### Author's contributions

AD implemented the first quickPIV version, which included spatial and FFT cross-correlation, signal-to-noise measurements, multi-pass and sub-pixel approximation. MP added efficient ZNCC and NSQECC cross-correlation, implemented the accuracy and performance evaluations, performed the analysis on *T. castaneum*, prepared the figures for this paper, and maintains the current version of the package. FK generated the *T. castaneum* data sets (i and ii) analyzed in this paper. FS oversaw the recording of the *T. castaneum* embryos and provided the knowledge of the developmental processes of *T. castaneum* to contrast the PIV-computed vector fields. ES provided the microscopy setup and valuable insights on LSFM technology and *T. castaneum* development. FM conceptualised and supervised the implementation, as well as the performed analyses, and contributed in the figure preparation. MP, AD, FS and FM wrote the manuscript. FK and ES contributed to the preparation of the manuscript and proof-reading. All authors have read and approved the final manuscript.

### Funding

Open Access funding enabled and organized by Projekt DEAL. MP and FM acknowledge funding by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—414985841. AD received funding from the LOEWE

programme of the state of Hessen (DynaMem). FM is supported by the Giersch Foundation, Frankfurt am Main, and receives funding from the LOEWE programme of the state of Hessen (DynaMem, CMMS). FK received funding from the LOEWE programme of the state of Hessen (CMMS). EHKS was funded by the Cluster of Excellence—Frankfurt am Main for Macromolecular Complexes (CEF-MC, speaker Volker Dötsch) at the Buchmann Institute for Molecular Life Sciences (BMLS) at the Goethe Universität—Frankfurt am Main by the Deutsche Forschungsgemeinschaft (DFG, EXC 115). FS was funded by the Add-on Fellowship 2019 of the Joachim Herz Stiftung. EHKS and FS were funded by the Quantitative Structural Cell Biology Projects programme (Innovations- und Strukturentwicklungsinitiative ‘Spitze aus der Breite’) of the state of Hessen. The funding body did not play any role in the design of the study and collection, analysis, and interpretation of data and in writing the manuscript.

#### Availability of data and materials

The data sets used and/or analyzed during the current study can be accessed through the following Zenodo <https://doi.org/10.5281/zenodo.5504076>.

#### Declaration

#### Ethics approval and consent to participate

Not applicable.

#### Consent for publication

Not applicable

#### Software and code

The source code and instructions to run the evaluation scripts of quickPIV are available at Github: <https://github.com/Marc-3d/quickPIV>. The source code of the presented release of quickPIV, v1.0, can be accessed through: <https://zenodo.org/record/5530573#.YVGyMX1CTBU>.

#### Competing interests

The authors declare that they have no competing interests.

#### Author details

<sup>1</sup>Frankfurt Institute for Advanced Studies (FIAS) and Goethe Universität Frankfurt am Main, Ruth-Moufang-Straße 1, 60438 Frankfurt am Main, Germany. <sup>2</sup>Buchmann Institute for Molecular Life Sciences (BMLS), Max-von-Laue Straße 15, 60438 Frankfurt am Main, Germany. <sup>3</sup>Heidelberg University Hospital, Im Neuenheimer Feld 410, 69120 Heidelberg, Germany.

Received: 3 March 2021 Accepted: 15 October 2021

Published online: 04 December 2021

#### References

- Friedl P, Gilmour D. Collective cell migration in morphogenesis, regeneration and cancer. *Nat Rev Mol Cell Biol*. 2009;10(7):445–57. <https://doi.org/10.1038/nrm2720>.
- Chuai M, Hughes D, Weijer CJ. Collective epithelial and mesenchymal cell migration during gastrulation. *Curr Genomics*. 2012;13(4):267–77. <https://doi.org/10.2174/138920212800793357>.
- Szabó A, Mayor R. Mechanisms of neural crest migration. *Annu Rev Genet*. 2018;52(1):43–63. <https://doi.org/10.1146/annurev-genet-120417-031559>.
- Jiang J, Li L, He Y, Zhao M. Collective cell migration: implications for wound healing and cancer invasion. *Burns Trauma*. 2013;1(1):21. <https://doi.org/10.4103/2321-3868.113331>.
- Barriga EH, Franze K, Charras G, Mayor R. Tissue stiffening coordinates morphogenesis by triggering collective cell migration in vivo. *Nature*. 2018;554(7693):523–7. <https://doi.org/10.1038/nature25742>.
- Vedula SRK, Leong MC, Lai TL, Hersen P, Kabla AJ, Lim CT, Ladoux B. Emerging modes of collective cell migration induced by geometrical constraints. *Proc Natl Acad Sci*. 2012;109(32):12974–9. <https://doi.org/10.1073/pnas.1119313109>.
- Lin S-Z, Ye S, Xu G-K, Li B, Feng X-Q. Dynamic migration modes of collective cells. *Biophys J*. 2018;115(9):1826–35. <https://doi.org/10.1016/j.bpj.2018.09.010>.
- Santi PA. Light sheet fluorescence microscopy: a review. *J Histochem Cytochem*. 2011;59(2):129–38. <https://doi.org/10.1369/0022155410394857>.
- Power RM, Huiskens J. A guide to light-sheet fluorescence microscopy for multiscale imaging. *Nat Methods*. 2017;14(4):360–73. <https://doi.org/10.1038/nmeth.4224>.
- Reynaud E, Krzic U, Greger K, Stelzer E. Light sheet-based fluorescence microscopy: more dimensions, more photons, and less photodamage. *HFSP J*. 2008;2:266–75. <https://doi.org/10.2976/1.2974980>.
- Kaufmann A, Mickoleit M, Weber M, Huiskens J. Multilayer mounting enables long-term imaging of zebrafish development in a light sheet microscope. *Development*. 2012;139(17):3242–7. <https://doi.org/10.1242/dev.082586>.
- Huiskens J. Optical sectioning deep inside live embryos by selective plane illumination microscopy. *Science*. 2004;305(5686):1007–9. <https://doi.org/10.1126/science.1100035>.
- Voleti V, Patel KB, Li W, Campos CP, Bharadwaj S, Yu H, Ford C, Casper MJ, Yan RW, Liang W, Wen C, Kimura KD, Targoff KL, Hillman EMC. Real-time volumetric microscopy of in vivo dynamics and large-scale samples with SCAPE 2.0. *Nat Methods*. 2019;16(10):1054–62. <https://doi.org/10.1038/s41592-019-0579-4>.
- Truong TV, Holland DB, Madaan S, Andreev A, Keomane-Dizon K, Troll JV, Koo DES, McFall-Ngai MJ, Fraser SE. High-contrast, synchronous volumetric imaging with selective volume illumination microscopy. *Commun Biol*. 2020;3(1). <https://doi.org/10.1038/s42003-020-0787-6>.

15. ...Chen B-C, Legant WR, Wang K, Shao L, Milkie DE, Davidson MW, Janetopoulos C, Wu XS, Hammer JA, Liu Z, English BP, Mimori-Kiyosue Y, Romero DP, Ritter AT, Lippincott-Schwartz J, Fritz-Laylin L, Mullins RD, Mitchell DM, Bembenek JN, Reymann A-C, Böhme R, Grill SW, Wang JT, Seydoux G, Tulu US, Kiehart DP, Betzig E. Lattice light-sheet microscopy: imaging molecules to embryos at high spatiotemporal resolution. *Science*. 2014;346(6208):1257998. <https://doi.org/10.1126/science.1257998>.
16. Sveen JK. An introduction to MatPIV v. 1.6.1. University of Oslo, Department of Mathematics;2004.
17. Thielicke W, Stamhuis EJ. PIVlab - towards user-friendly, affordable and accurate digital particle image velocimetry in MATLAB. *Journal of Open Research Software*. 2014;2. <https://doi.org/10.5334/jors.bl>.
18. Liberzon A, Lasagna D, Aubert M, Bachant P, Käufer T, Jakirkham, Bauer A, Vodenicharski B, Dallas C, Borg J, Tomerast, Ranleu. OpenPIV/openpiv-python: OpenPIV - Python (v0.22.2) with a new extended search PIV grid option. Zenodo. 2020. <https://doi.org/10.5281/ZENODO.3930343>
19. JPIV. <https://eguvep.github.io/jpiv/>
20. Bezanson J, Karpinski S, Shah VB, Edelman A. Julia: A fast dynamic language for technical computing. *CoRR abs/1209.5145* (2012). arxiv 1209.5145
21. Strobl F, Klees S, Stelzer EHK. Light sheet-based fluorescence microscopy of living or fixed and stained tribolium castaneum embryos (122). 2017. <https://doi.org/10.3791/55629>
22. Strobl F, Stelzer EHK. A deterministic genotyping workflow reduces waste of transgenic individuals by two-thirds. *Sci Rep*. 2021;11(1). <https://doi.org/10.1038/s41598-021-94288-0>
23. Adrian R. Twenty years of particle image velocimetry. *Exp Fluids*. 2005;39:159–69. <https://doi.org/10.1007/s00348-005-0991-7>.
24. Gollin D, Brevis W, Bowman ET, Shepley P. Performance of PIV and PTV for granular flow measurements. *Granular Matter* 2017;19(3). <https://doi.org/10.1007/s10035-017-0730-9>.
25. Ferrari S. Image analysis techniques for the study of turbulent flows. *EPJ Web of Conferences*. 2017;143:01001. <https://doi.org/10.1051/epjconf/201714301001>.
26. Melling A. Tracer particles and seeding for particle image velocimetry. *Meas Sci Technol*. 1997;8(12):1406–16. <https://doi.org/10.1088/0957-0233/8/12/005>.
27. Keane RD, Adrian RJ. Theory of cross-correlation analysis of PIV images. *Appl Sci Res*. 1992;49(3):191–215. <https://doi.org/10.1007/bf00384623>.
28. Stichel D, Middleton AM, Müller BF, Depner S, Klingmüller U, Breuhahn K, Matthäus F. An individual-based model for collective cancer cell migration explains speed dynamics and phenotype variability in response to growth factors. *NPJ Syst Biol Appl*. 2017;3(1). <https://doi.org/10.1038/s41540-017-0006-3>.
29. Weiger MC, Vedham V, Stuelten CH, Shou K, Herrera M, Sato M, Losert W, Parent CA. Real-time motion analysis reveals cell directionality as an indicator of breast cancer progression. *PLoS ONE*. 2013;8(3):58859. <https://doi.org/10.1371/journal.pone.0058859>.
30. Müller B, Bovet M, Yin Y, Stichel D, Malz M, González-Vallinas M, Middleton A, Ehemann V, Schmitt J, Muley T, Meister M, Herpel E, Singer S, Warth A, Schirmacher P, Drasdo D, Matthäus F, Breuhahn K. Concomitant expression of far upstream element (FUSE) binding protein (FBP) interacting repressor (FIR) and its splice variants induce migration and invasion of non-small cell lung cancer (NSCLC) cells. *J Pathol*. 2015;237(3):390–401. <https://doi.org/10.1002/path.4588>.
31. Glover JD, Wells KL, Matthäus F, Painter KJ, Ho W, Riddell J, Johansson JA, Ford MJ, Jahoda CAB, Klika V, Mort RL, Headon DJ. Hierarchical patterning modes orchestrate hair follicle morphogenesis. *PLoS Biol*. 2017;15(7):2002117. <https://doi.org/10.1371/journal.pbio.2002117>.
32. Zhang Y, Xu G, Lee RM, Zhu Z, Wu J, Liao S, Zhang G, Sun Y, Mogilner A, Losert W, Pan T, Lin F, Xu Z, Zhao M. Collective cell migration has distinct directionality and speed dynamics. *Cell Mol Life Sci*. 2017;74(20):3841–50. <https://doi.org/10.1007/s00018-017-2553-6>.
33. Zickus V, Taylor JM. 3D + time blood flow mapping using spim-micropiv in the developing zebrafish heart. *Biomed Opt Express*. 2018;9(5):2418–35. <https://doi.org/10.1364/BOE.9.002418>.
34. Vennemann P, Lindken R, Hierck B, Westerweel J. Volumetric particle image velocimetry in the developing chicken heart. *J Biomech*. 2006;39. [https://doi.org/10.1016/S0021-9290\(06\)85562-4](https://doi.org/10.1016/S0021-9290(06)85562-4).
35. Cheng C-M, Chang Y-F, Wu C-M. Cross-correlation analysis for live-cell image trajectory. 2013;8911:89110. <https://doi.org/10.1117/12.2034840>.
36. Cornwell JA, Li J, Mahadevan S, Draper JS, Joun GL, Zoellner H, Asli NS, Harvey RP, Nordon RE. Trackpad: Software for semi-automated single-cell tracking and lineage annotation. *SoftwareX*. 2020;11:100440. <https://doi.org/10.1016/j.softx.2020.100440>.
37. Cooley JW, Lewis PAW, Welch PD. Historical notes on the fast fourier transform. *Proc IEEE*. 1967;55(10):1675–7. <https://doi.org/10.1109/proc.1967.5959>.
38. Frigo M, Johnson SG. The design and implementation of fftw3. *Proc IEEE*. 2005;93:216–31. <https://doi.org/10.1109/JPROC.2004.840301>.
39. Lewis JP. Fast normalized cross-correlation *Ind Light Magic*. 2001;10.
40. Bradski, G.: *The OpenCV Library*. Dr. Dobbs Journal of Software Tools. 2000.
41. Bastiaans R. Cross-correlation PIV; Theory. Faculty of Mechanical Engineering, Eindhoven: Implementation and Accuracy. Eindhoven University of Technology; 2000.
42. Bromiley P. Products and convolutions of gaussian distributions. 2003.
43. Scarano F. A super-resolution particle image velocimetry interrogation approach by means of velocity second derivatives correlation. *Meas Sci Technol*. 2004;15:475. <https://doi.org/10.1088/0957-0233/15/2/023>.
44. Roth G, Katz J. Five techniques for increasing the speed and accuracy of PIV interrogation. *Meas Sci Technol*. 2001;12:238. <https://doi.org/10.1088/0957-0233/12/3/302>.
45. Scarano F, Riethmuller ML. Iterative multigrid approach in PIV image processing with discrete window offset. *Exp Fluids*. 1999;26(6):513–23. <https://doi.org/10.1007/s003480050318>.
46. Masullo A, Theunissen R: On dealing with multiple correlation peaks in PIV. *Exp Fluids* 2018;59(5). <https://doi.org/10.1007/s00348-018-2542-z>.

47. Charonko JJ, Vlachos PP. Estimation of uncertainty bounds for individual particle image velocimetry measurements from cross-correlation peak ratio. *Meas Sci Technol*. 2013;24(6):065301. <https://doi.org/10.1088/0957-0233/24/6/065301>.
48. Xue Z, Charonko JJ, Vlachos PP. Particle image velocimetry correlation signal-to-noise ratio metrics and measurement uncertainty quantification. *Meas Sci Technol*. 2014;25(11):115301. <https://doi.org/10.1088/0957-0233/25/11/115301>.
49. Raffel M, Willert CE, Scarano F, Kähler CJ, Wereley ST, Kompenhans, J.: PIV Uncertainty and Measurement Accuracy, pp. 203–241. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-68852-7\\_6](https://doi.org/10.1007/978-3-319-68852-7_6).
50. Chen, J., Revels, J.: Robust benchmarking in noisy environments. CoRR **abs/1608.04295** (2016). arxiv1608.04295
51. Preibisch S, Amat F, Stamatakis E, Sarov M, Singer RH, Myers E, Tomancak P. Efficient bayesian-based multiview deconvolution. *Nat Methods*. 2014;11(6):645–8. <https://doi.org/10.1038/nmeth.2929>.
52. Thielicke, W.: The flapping flight of birds: Analysis and application. PhD thesis, University of Groningen (2014)
53. Michaelis D, Neal DR, Wieneke B. Peak-locking reduction for particle image velocimetry. *Meas Sci Technol*. 2016;27(10):104005. <https://doi.org/10.1088/0957-0233/27/10/104005>.
54. Nobach H, Bodenschatz E. Limitations of accuracy in PIV due to individual variations of particle image intensities. *Exp Fluids*. 2009;47(1):27–38. <https://doi.org/10.1007/s00348-009-0627-4>.
55. Merzkirch W, Gui L. A comparative study of the MQD method and several correlation-based PIV evaluation algorithms. *Exp Fluids*. 2000;28(1):36–44. <https://doi.org/10.1007/s003480050005>.
56. Forliti DJ, Strykowski PJ, Debatin K. Bias and precision errors of digital particle image velocimetry. *Exp Fluids*. 2000;28(5):436–47. <https://doi.org/10.1007/s003480050403>.
57. Handel K, Grünfelder CG, Roth S, Sander K. Tribolium embryogenesis: a SEM study of cell shapes and movements from blastoderm to serosal closure. *Dev Genes Evol*. 2000;210(4):167–79. <https://doi.org/10.1007/s004270050301>.
58. Münster S, Jain A, Mietke A, Pavlopoulos A, Grill SW, Tomancak P. Attachment of the blastoderm to the vitelline envelope affects gastrulation of insects. *Nature*. 2019;568(7752):395–9. <https://doi.org/10.1038/s41586-019-1044-3>.
59. Benton MA, Akam M, Pavlopoulos A. Cell and tissue dynamics during tribolium embryogenesis revealed by versatile fluorescence labeling approaches. *Development*. 2013;140(15):3210–20. <https://doi.org/10.1242/dev.096271>.
60. Jain, A., Ulman, V., Mukherjee, A., Prakash, M., Cuenca, M.B., Pimpale, L.G., Münster, S., Haase, R., Panfilio, K.A., Jug, F., Grill, S.W., Tomancak, P., Pavlopoulos, A.: Regionalized tissue fluidization is required for epithelial gap closure during insect gastrulation. *Nature Communications* **11**(1) (2020). <https://doi.org/10.1038/s41467-020-19356-x>
61. Liu Y, Zou Q, Luo S. GPU Accelerated Fourier Cross Correlation Computation and Its Application in Template Matching. 2011;163:484–91. [https://doi.org/10.1007/978-3-642-25002-6\\_68](https://doi.org/10.1007/978-3-642-25002-6_68).

## Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ready to submit your research? Choose BMC and benefit from:

- fast, convenient online submission
- thorough peer review by experienced researchers in your field
- rapid publication on acceptance
- support for research data, including large and complex data types
- gold Open Access which fosters wider collaboration and increased citations
- maximum visibility for your research: over 100M website views per year

At BMC, research is always in progress.

Learn more [biomedcentral.com/submissions](https://biomedcentral.com/submissions)

